Problem 1: (LSH for Angular Similarity)

For any vectors $x, y \in \mathbb{R}^d$, the angular distance is the angle (in radians) between the two vectors—formally, $d_\theta(x, y) = \cos^{-1}\left(\frac{x \cdot y}{||x||_2 ||y||_2}\right)$ (where $\cos^{-1}(\cdot)$ returns the principle angle, i.e., angles in $[0, \pi]$). The (normalized) angular similarity is given by $s_\theta(x, y) = 1 - d_\theta(x, y)/\pi$.

We now want to construct a LSH for the angular similarity metric. Consider the following family of hash functions: we first choose a random unit vector $\sigma$ (i.e., $\sigma \in \mathbb{R}^d$ with $||\sigma||_2 = 1$), and for any vector $x$, define $h_\sigma(x) = \text{sgn}(x . \sigma)$ (i.e., the sign of the dot product of $x$ and $\sigma$). Argue that for any $x, y \in \mathbb{R}^d$, we have:

$$\mathbb{P}[h_\sigma(x) = h_\sigma(y)] = s_\theta(x, y)$$

Hint: For any pair $x$ and $y$ in $\mathbb{R}^d$, there is a unique plane passing through the origin containing $x$ and $y$—convince yourself that $d_\theta(x, y)$ is precisely the angle between $x$ and $y$ in this plane. Also, given any vector $\sigma$, its dot product with $x$ and $y$ only depends on the projection of $\sigma$ on this plane. Now what can you say about the signs of the dot products of $x$ and $y$ with a random unit vector?

Solution: Vectors $x$ and $y$ always define a plane, and the angle between them is measured in this plane. Figure (1) is a “top-view” of the plane containing $x$ and $y$.

![Figure 1: Two vectors make an angle $\theta$](image)

Suppose we pick a hyperplane through the origin. This hyperplane intersects the plane of $x$ and $y$ in a line. Figure (1) suggests two possible hyperplanes, one whose intersection is the dashed line and the other’s intersection is the dotted line. To pick a random hyperplane, we actually pick the normal vector to the hyperplane, say $\sigma$. The hyperplane is then the set of points whose dot product with $\sigma$ is 0.

First, consider a vector $\sigma$ that is normal to the hyperplane whose projection is represented by the dashed line in Fig. (1); that is, $x$ and $y$ are on different sides of the hyperplane. Then the dot products $\sigma . x$ and $\sigma . y$ will have different signs. If we assume, for instance, that $\sigma$ is a vector whose
projection onto the plane of \(x\) and \(y\) is above the dashed line in Fig. (1), then \(\sigma.x\) is positive, while \(\sigma.y\) is negative. The normal vector \(\sigma\) instead might extend in the opposite direction, below the dashed line. In that case \(\sigma.x\) is negative and \(\sigma.y\) is positive, but the signs are still different.

On the other hand, the randomly chosen vector \(\sigma\) could be normal to a hyperplane like the dotted line in Fig. (1). In that case, both \(\sigma.x\) and \(\sigma.y\) have the same sign. If the projection of \(\sigma\) extends to the right, then both dot products are positive, while if \(\sigma\) extends to the left, then both are negative.

What is the probability that the randomly chosen vector is normal to a hyperplane that looks like the dashed line rather than the dotted line? All angles for the line that is the intersection of the random hyperplane and the plane of \(x\) and \(y\) are equally likely. Thus, the hyperplane will look like the dashed line with probability \(\theta/\pi\) and will look like the dotted line otherwise.

**Problem 2: (Choosing LSH Parameters for Nearest Neighbors)**

An important routine in many clustering/machine learning algorithms is the \((c, R)\)-Nearest-Neighbors (or \((c, R)\)-NN) problem: given a set of \(n\) points \(V\) and a distance metric \(d\), we want to store \(V\) in order to support the following query:

*Given a query point \(q\), if there exists \(x \in V\) such that \(d(x, q) \leq R\) then, with probability at least \(1 - \delta\), we must output a point \(x' \in V\), such that \(d(x', q) \leq cR\).*

We now show how to solve this problem using LSH. Assume that we are given a \((R, cR, p_1, p_2)\)-sensitive hash family \(H\). As in class, we can amplify the probabilities by first taking the AND of \(r\) such hash functions to get a new family \(H_{AND}\); next, we can take the OR of \(b\) hash functions from \(H_{AND}\) to get another family \(H_{OR-AND}\).

Given the set \(V\), we hash each element using a single hash function \(g\) from \(H_{OR-AND}\) (which corresponds to \(b \times r\) hash functions from \(H\)). Now given a query point \(q\), we hash \(q\) using our cascaded hash-function \(g\), and find all \(y \in V\) such that \(g(y) = g(q)\) – let this set be denoted \(Y_q\). Finally, we can check \(d(q, y)\) for each \(y\) in \(Y_q\), and return those \(y\) for whom \(d(q, y) < cR\).

**Part (a)**

If there exists \(x \in V\) such that \(d(x, q) \leq R\) then, argue that we output \(x\) with probability \(1 - (1 - p_1)^b\). On the other hand, also show that the expected number of false positives (i.e., points \(x' \in V\) such that \(d(x', q) > cR\)) that we consider per hash function in \(H_{AND}\) is at most \(np_2^2\).

**Solution:** From the definition of a \((R, cR, p_1, p_2)\)-sensitive hash family, we know that for any \(x \in V\) such that \(d(x, q) \leq R\), the probability that there is a collision is at least \(p_1\) – hence the probability that all the hash functions do not collide is \(1 - p_1^b\). Now since we are taking the OR of \(b\) such hash functions from the family \(H_{AND}\), the probability that none of them output \(x\) is at most \(1 - (1 - p_1^b)^b\).

On the other hand, for any \(x \in V\) such that \(d(x, q) \geq cR\), we know that for any composite hash function in \(H_{AND}\), a false collision occurs with probability at most \(p_2^2\). Now to bound the expected

---

\(^1\)Recall in class we defined a \((d_1, d_2, p_1, p_2)\)-sensitive hash family – for convenience, we are setting the distances to \(R\) and \(cR\).
number of false positives, note that the number of elements \( x \) such that \( d(x,q) \geq cR \) is bounded by \(|V| = n\) – thus the expected number of false positives is at most \( np_2^\rho\).

**Part (b)**

Note that since we check for false positives, we never output one – however, we have \( O(1) \) runtime cost for each false positive (to check its distance). Choose \( r \) to ensure that the expected number of false-positives per hash function in \( H_{AND} \) is 1. Using this choice of \( r \), show that for the guarantee we desire for the \((c,R)-NN\) problem, we need to choose \( b = n^\rho \ln(1/\delta) \), where \( \rho = \frac{\ln(1/p_1)}{\ln(1/p_2)} \).

**Solution:** To ensure that on average we have at most one false positive, we can choose \( r \) such that \( np_2^\rho = 1 \) – thus \( r = \ln n / \ln(1/p_2) \) – thus \( (1/p_1)^r = \exp(\ln(1/p_1) \ln n / \ln(1/p_2)) = n^\rho \). Now suppose we choose \( b = n^\rho \ln(1/\delta) \) – then we have:

\[
1 - (1 - p_1^r)^b = 1 - \left(1 - \frac{1}{n^\rho}\right)^{n^\rho \ln(1/\delta)} \geq 1 - e^{-\ln(1/\delta)} = 1 - \delta,
\]

where we have used \((1 - x) < e^{-x}\). Thus, we have that for this choice of \( b \) and \( r \), any \( x \in V \) such that \( d(x,q) \geq cR \) is returned with probability at least \( 1 - \delta \), while we return on average one \( x' \in V \) such that \( d(x',q) \leq cR \).

**Problem 3: (More on the Morris’ Counter)**

Recall in class we saw the basic Morris counter, wherein we initiated the counter to 1 when one item arrived, and upon each subsequent arrival, incremented the counter with probability \( 1/2^X \). We also showed that after \( n \) items have arrived, \( E[2^X] = n + 1 \).

**Part (a)**

Prove that the variance of the counter is given by:

\[
Var(2^X_n) = \frac{n^2 - n}{2}
\]

Using this, find the probability that the average of \( k \) Morris counters is less than \( n + 1 - \epsilon n \) after \( n \) items have passed.

*Hint: Use induction for \( E[2^X] \).*

**Solution:** Let counter’s state after seeing \( n \) items be \( X_n \) – recall that we showed in class that \( E[2^X_n] = n + 1 \). Since, we want to prove that \( Var(2^X_n) = \frac{n^2 - n}{2} \), this is equivalent to showing:

\[
E[2^{2X_n}] = Var(2^X_n) + (E[2^X_n])^2 = \frac{n^2 - n}{2} + (n + 1)^2 = \frac{3}{2} n^2 + \frac{3}{2} n + 1.
\]
We will now show this by induction. Clearly for $X_0 = 1$, we have $E[2^{2^{-1}}] = \frac{3}{2}(1)^2 + \frac{3}{2}(1) + 1 = 4$. For the inductive step, we have:

$$E[2^{2X_n}] = \sum_{j=0}^{\infty} P(2^{X_{n-1}} = j) \cdot E[2^{2X_n} | 2^{X_{n-1}} = j]$$

$$= \sum_{j=0}^{\infty} P(2^{X_{n-1}} = j) \cdot \left[ \frac{1}{j} \cdot 4j^2 + \left( 1 - \frac{1}{j} \right) \cdot j^2 \right]$$

$$= \sum_{j=0}^{\infty} P(2^{X_{n-1}} = j) \cdot (j^2 + 3j)$$

$$= E[2^{2X_{n-1}}] + 3 \cdot E[2^{X_{n-1}}] = \frac{3}{2} (n-1)^2 + \frac{3}{2} (n-1) + 1 + 3n$$

$$= \frac{3}{2} n^2 + \frac{3}{2} n + 1.$$ 

Now, assume we have $k$ Morris counters $X_1, \ldots, X_k$, and $Z = \frac{1}{k} \sum_{j=1}^{k} 2^X_j$. Then, by independence:

$$Var(Z) = \frac{1}{k^2} Var \left( \sum_{j=1}^{k} 2^X_j \right) = \frac{n^2 - n}{2k}.$$ 

By Chebyshev’s inequality:

$$P(Z < n + 1 - \epsilon n) \leq P(|Z - (n + 1)| > \epsilon n) \leq \frac{Var(Z)}{\epsilon n^2} = \frac{n - 1}{2k \epsilon^2}.$$

**Part (b)**

Next, suppose we modify the counter as follows: we still initialize counter $Y$ to 1 when the first item arrives, but on every subsequent arrival, we increment the counter by $1$ with probability $1/(1+a)^Y$, for some $a > 0$. Let $Y_n$ be the counter-state after $n$ items have arrived – choose constants $b, c$ such that $b \cdot (1 + a)^{Y_n} + c$ is an unbiased estimator for the number of items (i.e., $E[b \cdot (1 + a)^{Y_n} + c] = n$).

**Solution:** First, since $Y_0 = 0$, hence $E[(1 + a)^{Y_0}] = 1$. Now as in the previous analysis, we have:

$$E[(1 + a)^{Y_n}] = \sum_{j=0}^{\infty} P(Y_{n-1} = j) E[(1 + a)^{Y_n} | Y_{n-1} = j]$$

$$= \sum_{j=0}^{\infty} P(Y_{n-1} = j) \left( \frac{1}{(1+a)^j}(1+a)^{j+1} + \left( 1 - \frac{1}{(1+a)^j} \right) (1+a)^j \right)$$

$$= E[(1 + a)^{Y_{n-1}}] + a.$$ 

Thus, we have that $E[(1 + a)^{Y_n}] = 1 + na$. Thus, if we choose $b = 1/a, c = -1/a$, we get:

$$E[b \cdot (1 + a)^{Y_n} + c] = \frac{1 + na}{a} - \frac{1}{a} = n.$$
Part (c) (OPTIONAL)

Now suppose you are restricted to use a single Morris counter, but can choose \( a \) as above. Find the variance of the estimator, and using Chebyshev, find the required \( a \) to ensure that the estimate is within \( n \pm \epsilon n \) with probability at least \( 1 - \delta \). What is the expected storage required by this counter?

Problem 4: (Dyadic Partitions and the Count-Min Sketch)

In this problem, we modify the Count-Min sketch to give estimates for range queries and heavy-hitters. For this, we first need an additional definition. For convenience, assume \( n = 2^k \); the dyadic partitions of the set \([n]\) are defined as follows:

\[
I_0 = \{\{1\}, \{2\}, \ldots, \{n\}\}
\]

\[
I_1 = \{\{1, 2\}, \{3, 4\}, \ldots, \{n - 1, n\}\}
\]

\[
I_2 = \{\{1, 2, 3, 4\}, \{5, 6, 7, 8\}, \ldots, \{n - 3, n - 2, n - 1, n\}\}
\]

\[
\vdots
\]

\[
I_k = \{\{1, 2, \ldots, n\}\}
\]

Part (a)

Let \( I = I_0 \cup I_1 \cup \ldots \cup I_k \) be the set of all dyadic intervals. Show that \( |I| \leq 2n \). Moreover, show that any interval \([a, b]\) can be written as a disjoint union of at most \( 2 \log_2 n \) sets from \( I \). (For example, for \( n = 16 = 2^4 \), the set \([6, 15]\) can be written as \( \{6\} \cup \{7, 8\} \cup \{9, 10, 11, 12\} \cup \{13, 14\} \cup \{15\} \), which is less than \( 2 \times 4 = 8 \) sets.)

Solution: By definition of the dyadic intervals, we have that for any \( i \in \{0, 1, \ldots, k\} \), we have that \( |I_i| = n/2^i \). Thus the number of dyadic intervals is given by \( |I| = \sum_{i=0}^{k} n2^{-i} \leq n \sum_{i=0}^{\infty} 2^{-i} = 2n \).

For the second claim, we can use induction on \( k = \log_2 n \). The base case of \( k = 1 \) (\( n = 2 \)) is easy to check. Now suppose that for \( k - 1 \) we have that any sub-interval can be represented as a disjoint union of \( 2(k - 1) \) dyadic intervals. Now given a sub-interval \([a, b]\) = \( \{a, a+1, \ldots, b\} \) of \([2^k]\), if either \( a > 2^{k-1} = n/2 \) or \( b \leq 2^{k-1} = n/2 \), then we are done by the inductive hypothesis. To complete the proof, we need to show that if \( a < n/2 < b \), then we can write \([a, b]\) as a disjoint union of \( 2k \) dyadic intervals.

We first show that for any \( a \in [2^{k-1}] \), we can write the set \( \{1, 2, \ldots, a\} \) as a disjoint union of at most \( k - 1 \) dyadic intervals. This again we can see by induction. Again the base case is easy to check. Moreover, for any \( a \in [2^{k-1}] \), we have two cases: i) if \( a \leq 2^{k-2} \), then by induction we need \( \leq k - 2 \) intervals, and ii) if \( a > 2^{k-2} \), then by induction we need \( \leq 1 + k - 2 = k - 1 \) intervals.

Now by symmetry, we also have that for any interval \([b, b+1, \ldots, 2^{k-1}]\) we can write it as a disjoint union of at most \( k - 1 \) dyadic intervals (just reverse the sets!). Returning to the main proof, given \( a < n/2 < b \), we can write \([a, b] = [a, n/2] \cup [n/2, b]\) – from the above claims, each can be written as a disjoint union of \( k - 1 \) dyadic intervals, and hence we have \([a, b]\) can be written using \( 2k - 2 < 2k \) intervals, which completes the proof.
Part (b)

In class, given a stream of $m$ elements, we saw how to construct a Count-Min sketch for the frequencies of items $i \in [n]$, and how to use it for point queries (i.e., to estimate $f_i$ for some $i \in [n]$). We now extend this to range queries – estimating $F_{[a,b]} = \sum_{i=a}^{b} f_i$ for given $a, b$.

Note first that the basic Count-Min sketch can be interpreted as constructing a sketch for frequencies of set-membership for the sets in $I_0$. We have also seen how to make hash functions for general set-membership (for example, the Bloom filter!) – we can thus extend the Count-Min sketch to include an estimate for the frequencies of all the dyadic intervals. Using this new sketch, show that for a given range query $[a,b]$, we can use a Count-Min sketch with $R = \log(1/\delta)$ rows and $B = 2/\epsilon$ columns to get an estimate $F_{[a,b]}$ satisfying:

$$\mathbb{P} \left[ F_{[a,b]} < \sum_{i \in [a,b]} f_i + 2m\epsilon \log^2 n \right] \geq 1 - \delta$$

Solution: (Note: Correction in the above expression - the RHS of the bound on $F_{[a,b]}$ should be $\log^2 n$, not $\log n$ as was given in the problem.)

First, note that the size of the Count-Min data-structure did not depend on the number of elements $n$ – thus, we can adapt the Count-Min sketch to store counts $F_I$ for all sets $I \in \mathcal{I}$. Note however that each $i \in [n]$ belongs to $\log_2 n$ dyadic intervals – thus instead of counting $m$ items, we are counting $m \log_2 n$ items.

Next, from the previous part, we know that any interval $[a, b]$ can be written as the disjoint union of $\leq 2 \log_2 n$ dyadic intervals – let us denote this set as $\mathcal{I}_{[a,b]}$. Thus we have $F_{[a,b]} = \sum_{I \in \mathcal{I}_{[a,b]}} F_I$. Moreover, note that each $F_I \leq m$.

Now from the performance bounds for the Count-Min sketch (with $R = \log(1/\delta)$ rows and $B = 2/\epsilon$ columns, and $m \log_2 n$ items in the stream) we saw in class, we know that for any $I \in \mathcal{I}$, we have:

$$\mathbb{P} \left[ F_I < \sum_{i \in [a,b]} f_i + (m \log_2 n)\epsilon \right] \geq 1 - \delta$$

Since we are adding $2 \log_2 n$ such counts for $F_{[a,b]}$, we get that:

$$\mathbb{P} \left[ F_{[a,b]} < \sum_{i \in [a,b]} f_i + 2m\epsilon \log^2 n \right] \geq 1 - \delta$$

Part (c)

The $\phi$-heavy-hitters (or $\phi$-HH) query is defined as follows:

Given stream $\{x_1, x_2, \ldots, x_m\}$ with $x_i \in [n]$, and some constant $\phi \in [0,1]$, we want to output a subset $L \subset [n]$ such that, with probability at least $1 - \delta$, $L$ contains all $i \in [n]$ such that $f_i \geq \phi m$, and moreover, every $i \in L$ satisfies $f_i \geq \phi m/2$. 
We now adapt the above sketch for the $\phi$-HH problem. First, using the union bound, argue that if we choose $\delta = \gamma/2n$, then we have that for all dyadic intervals $I \in \mathcal{I}$, we have that the frequency estimate $F_I$ obeys: $P \left[ F_I < \sum_{i \in I} f_i + m\epsilon \right] \geq 1 - \gamma$. Thus, argue that if we use $\epsilon < \phi/2$, then the set of all $i \in [n]$ such that $F_{\{i\}} > \phi m$ is a solution to the $\phi$-HH problem.

Solution: (Note: There was a typo in the probability bound – it should be $1 - \gamma$, not $1 - \delta$.) Suppose we choose $\delta = \gamma/2n$. Then, from the union bound, we have that:

$$P \left[ \bigcup_{I \in \mathcal{I}} \left\{ F_I > \sum_{i \in I} f_i + m\epsilon \right\} \right] \leq 2n P \left[ \left\{ F_I > \sum_{i \in I} f_i + m\epsilon \right\} \right] \leq 2n\delta = \gamma$$

Now, if we use $\epsilon = \phi/2$, then we have that:

- For any $i \in [n]$ such that $f_i \geq \phi m$, then $F_{\{i\}}$ is also $\geq \phi m$ (recall that the Count-Min sketch always overestimates frequencies!).
- For any $i \in [n]$ such that $f_i < \phi m/2$, then with probability $\geq 1 - 2\gamma$, we have that $F_{\{i\}}$ is also $\leq \phi m$.

Thus, if we use $\epsilon < \phi/2$, then the set of all $i \in [n]$ such that $F_{\{i\}} > \phi m$ is a solution to the $\phi$-HH problem (with $\gamma$ instead of $\delta$ as the probability bound).

Part (d)

Note though that the brute force way to find all $i \in [n]$ such that $F_{\{i\}} \geq \phi m$ requires $n$ point queries. Briefly argue how you can use the frequency estimates $F_I$ for the dyadic intervals to find the same using $O(\log n/\phi)$ queries.

Hint: Consider a binary tree defined by the dyadic intervals, with the root as $I_{\log_2 n} = \{[n]\}$, and the leaves as $I_0 = \{\{1\}, \{2\}, \ldots, \{n\}\}$. Argue that for every heavy-hitter node $i$, every parent node in the tree has $F_I > \phi m$. Also, at any level $j$, how many sets $I \in \mathcal{I}_j$ can have $F_I > \phi m$?

Solution: The main idea is that if $f_i > \phi m$, then $f_I > \phi m$ for any dyadic interval $I$ that contains $i$. Thus, we can start from the top of the tree of dyadic intervals, and at each stage, only expand dyadic intervals $I$ such that $F_I > \phi m$. Now note that at any level of the tree, the dyadic intervals form a partition of $[n]$ – thus their frequencies must add up to $m$. By a counting argument, we see that the number of intervals $I \in \mathcal{I}_j, i \in \{0, 1, \ldots, \log_2 n\}$ such that $f_I > \phi m$ is $O(1/\phi)$ (moreover, with high probability, the number of intervals such that $F_I > \phi m$ is $O(1/\phi)$). Finally, the depth of the tree is $\log n$. Thus, in $O(\log n/\phi)$ time, we can find all $i$ such that $F_{\{i\}} > \phi m$. 

7